Geochemistry Geophysics Geosystems G3

Article
Volume 12, Number 4
1 April 2011
Q04006, doi:10.1029/2010GC003425
ISSN: 1525-2027

Published by AGU and the Geochemical Society

Implementation and application of adaptive mesh refinement for thermochemical mantle convection studies

Wei Leng
Department of Physics, University of Colorado at Boulder, Boulder, Colorado 80309, USA

Now at Seismological Laboratory, California Institute of Technology, 1200 E. California Boulevard, 252-21, Pasadena, California 91125, USA (leng@caltech.edu)

Shijie Zhong
Department of Physics, University of Colorado at Boulder, Boulder, Colorado 80309, USA

[1] Numerical modeling of mantle convection is challenging. Owing to the multiscale nature of mantle dynamics, high resolution is often required in localized regions, with coarser resolution being sufficient elsewhere. When investigating thermochemical mantle convection, high resolution is required to resolve sharp and often discontinuous boundaries between distinct chemical components. In this paper, we present a 2-D finite element code with adaptive mesh refinement techniques for simulating compressible thermochemical mantle convection. By comparing model predictions with a range of analytical and previously published benchmark solutions, we demonstrate the accuracy of our code. By refining and coarsening the mesh according to certain criteria and dynamically adjusting the number of particles in each element, our code can simulate such problems efficiently, dramatically reducing the computational requirements (in terms of memory and CPU time) when compared to a fixed, uniform mesh simulation. The resolving capabilities of the technique are further highlighted by examining plume-induced entrainment in a thermochemical mantle convection simulation.

Components: 10,000 words, 12 figures, 3 tables.

Keywords: adaptive mesh refinement; mantle convection; thermochemical.

Index Terms: 0560 Computational Geophysics: Numerical solutions (4255); 0550 Computational Geophysics: Model verification and validation; 8121 Tectonophysics: Dynamics: convection currents, and mantle plumes.

Received 2 November 2010; **Revised** 4 March 2011; **Accepted** 9 March 2011; **Published** 15 April 2011.

Leng, W., and S. Zhong (2011), Implementation and application of adaptive mesh refinement for thermochemical mantle convection studies, *Geochem. Geophys. Geosyst.*, *12*, Q04006, doi:10.1029/2010GC003425.

1. Introduction

1.1. Motivation

[2] Numerical modeling, in combination with observations from geology, seismology, and geochemistry, is a very powerful tool to study the geodynamics of the Earth's mantle. Three-dimensional models in Cartesian and spherical geometries with variable viscosities have been developed with advances of computational capability [e.g., *Tackley*, 1996; *Bunge et al.*, 1997; *Zhong et al.*, 2000]. However, the resolution problem arises in mantle convection modeling when more realistic physics is included, such as when using very high Rayleigh numbers [e.g., *Zhong*, 2005], or including chemical piles at the bottom of the Earth's mantle for thermochemical convection [e.g., *van Keken et al.*, 1997;

Kellogg et al., 1999; *McNamara and Zhong*, 2004, 2005]. In order to resolve fine structures in these models, very high resolution is needed which may exceed the memory storage and computational capability for current computer systems. Particularly, high resolution (more elements) is a crucial requirement in thermochemical mantle convection studies in order to resolve fine chemical boundaries and to avoid artificial chemical diffusion [e.g., *van Keken et al.*, 1997].

[3] Many mantle convection codes like Stag3D [*Tackley*, 1996] or Citcom [*Moresi et al.*, 1996; *Zhong et al.*, 2000] allow local mesh refinement before the starting of calculations, such as to resolve the thin top and bottom thermal boundary layers in thermal convection by presetting finer meshes for the boundary layers. However, since the meshes used in Stag3D and Citcom have assumed connectivity which must remain fixed in time, they cannot resolve fine structures which develop over time (e.g., thermal plumes in thermal convection or chemical boundaries in thermochemical mantle convection).

[4] A technique, termed adaptive mesh refinement (AMR), allows us to refine meshes locally wherever higher resolution is required (e.g., in regions with high temperature gradients or large viscosity contrasts), while leaving other regions with relatively coarse resolution [*Löhner et al.*, 1985; *Peraire et al.*, 1987; *Albers*, 2000; *Tu et al.*, 2005, 2006; *Davies et al.*, 2007, 2008; *Burstedde et al.*, 2008; *Stadler et al.*, 2010]. More importantly, AMR enables us to deal with dynamical adjustment of mesh distribution, which is extremely important for tracing and resolving thermal plumes or chemical boundaries in thermochemical mantle convection studies.

[5] For a mantle underlain by a dense chemical layer, the dense material is continuously entrained by upwelling mantle plumes, provided the excess chemical density is low enough. Estimating how fast the dense materials are entrained from the deep mantle into the upper mantle is important for understanding the evolution history of the Earth's mantle and the distribution of mantle chemical heterogeneities [*Sleep*, 1988; *Davaille*, 1999; *Jellinek and Manga*, 2002; *Gonnermann et al.*, 2002; *Zhong and Hager*, 2003; *Huang*, 2008]. The entrainment rate of a dense layer by mantle plumes has been studied analytically [*Sleep*, 1988], experimentally [*Davaille*, 1999; *Jellinek and Manga*, 2002; *Gonnermann et al.*, 2002] and numerically [*Zhong and Hager*, 2003; *Huang*, 2008]. However, previous numerical models are confined by a resolution limit which is imposed by the computation time and memory requirements. Thus, a specific mesh setup has to be used in order to resolve the chemical boundaries, i.e., presetting plume positions and the mesh distribution [*Zhong and Hager*, 2003; *Huang*, 2008]. With AMR techniques, it becomes possible to locally modify the mesh resolution to resolve the evolving chemical boundaries and to study the entrainment problem in general thermochemical convection.

1.2. Previous Work

[6] AMR has been successfully implemented into mantle convection models [*Albers*, 2000; *Davies et al.*, 2007, 2008; *Stadler et al.*, 2010]. *Albers* [2000] was able to treat viscosity variations up to 10 orders of magnitude with a local mesh refinement technique in thermal convection models. Based on the octree data structure and AMR techniques, *Stadler et al.* [2010] were able to simulate plate tectonics with individual plate margins resolved down to a scale of 1 km in a 3-D global mantle convection model, but their models are also for purely thermal convection. *Davies et al.* [2007] implemented fully unstructured AMR, for both thermal and thermochemical convection. Comparison of their model predictions with benchmark results demonstrates the accuracy and efficiency of such techniques for purely thermal convection. However, for thermochemical models, Davies et al. used a field method for simulating the chemical composition. Owing to the large numerical diffusion associated with such schemes [*van Keken et al.*, 1997], their results failed to match the benchmark values, even with the localized high resolution afforded by AMR. AMR alone therefore, is not sufficient to ensure accurate thermochemical simulations and a specialized method for dealing with the compositional field is required. It is thus important to determine whether or not AMR can be successfully linked with another method for tracking chemical heterogeneities, which is one goal of the current paper.

[7] There are several ways to simulate different chemical compositions in thermochemical convection, including the field method, marker chain method, tracer method and particle level set method. In the field method, the compositional field is described by a continuous field (similar to temperature) and modeled by an advection-diffusion equation with a small compositional diffusivity [*van Keken et al.*, 1997]. With the continuous field method, the inaccurate flux approximation close to

the discontinuous chemical boundaries introduces spurious numerical undershoots and overshoots that produce dispersion errors. These dispersion errors may lead to numerical instability, and a filter scheme is required to remove these errors [*Lenardic and Kaula*, 1993]. An intrinsic problem for the field method is the severe numerical diffusion [*van Keken et al.*, 1997], for which high resolution provided with AMR techniques helps little [*Davies et al.*, 2007]. In the marker chain method, the interface between two different compositions is discretized with a series of markers that are advected to track the evolution of the chemical interface. The marker chain method is good at avoiding numerical diffusion. But when the chemical composition region is stretched to a large extent, this method suffers from the exponential growth of the chain [*van Keken et al.*, 1997]. The marker chain method is also difficult to generalize to 3-D models.

[8] Another way to simulate different chemical compositions is through the tracer method [e.g., *van Keken et al.*, 1997; *Tackley and King*, 2003]. In this method, the chemical compositions are represented by a large number of passive particles that advect with the flow field. The tracer method is the most popular one used in the geodynamics community, not only because it largely avoids numerical diffusion, but because its implementation is quite simple and straightforward. Recently, a particle level set method is introduced for capturing chemical interface [*Enright et al.*, 2002; *Hieber and Koumoutsakos*, 2005] and is implemented for studying geophysical problems [*Braun et al.*, 2008; *Samuel and Evonuk*, 2010]. This method puts particles in the vicinity of chemical interfaces. The particles define an implicit level set function from which the interface location can be extracted via a contouring algorithm. It is proposed that the particle level set method performs faster than the normal tracer method and is good at tracking sharp discontinuities [*Samuel and Evonuk*, 2010]. But due to the introduction of the level set function and the required reinitialization processes, the implementation of the particle level set method may not be as straightforward as the tracer method.

1.3. This Study

[9] Here we focus on how the tracer method can be implemented together with AMR. The classical tracer methods only use one type of tracers to represent the different chemical composition [*van Keken et al.*, 1997, and references therein]. Even if particles are uniformly spaced initially, the density of particles can vary significantly in space as convection evolves with time, which leads to the particle distribution problem. In some elements, there may be too few, or no particles such that the chemical composition for these elements cannot be appropriately represented, while for other elements, there may be excessive number of particles which are not necessary. The distribution problem can be improved by a tracer ratio method, i.e., having two different "dense" and "regular" particles with composition being equal to the local fraction of "dense" particle [e.g., *Tackley and King*, 2003]. However, with the adaptive mesh refinement method where meshes are dynamically refined and coarsened, the number of particles in elements may decrease/increase dramatically due to mesh refinement/coarsening. Therefore the particle distribution problem becomes much worse even with the tracer ratio method and needs to be dealt with care.

[10] Based on the finite element method, and inspired by AMR method in the studies by *Burstedde et al.* [2008] and *Stadler et al.* [2010], we develop a compressible thermochemical mantle convection code with the adaptive mesh refinement techniques in 2-D Cartesian geometry. We first present numerical techniques for implementing the adaptive mesh refinement into a compressible mantle convection model. After that, we demonstrate how to incorporate the tracer ratio method together with the AMR techniques. Then, the numerical results from the AMR code are compared with analytic solutions and previously published benchmark results for Stokes flow, thermal convection, single vortex flow and Rayleigh-Taylor instability problems. To finish, we present an application of the AMR code by analyzing the chemical entrainment rate for thermochemical convection and demonstrate the powerful resolving capability of the code.

2. Numerical Techniques for Adaptive Mesh Refinement

2.1. The Governing Equations for Thermochemical Convection in the Earth's Mantle

2.1.1. Compressible Mantle Convection

[11] The nondimensional governing equations for compressible mantle convection include the con-

servation equations of mass, momentum and energy [e.g., *Leng and Zhong*, 2008]

$$(\rho_r u_i)_{,i} = 0, \tag{1}$$

$$-p_{,j}\delta_{ij} + \tau_{ij,j} + [\rho_r Ra(T - T_r) - p\gamma]\delta_{i2} = 0, \tag{2}$$

$$\rho_r \dot{T} + u_i T_{,i} + \rho_r \overline{D} u_2 (T + T_s) = (\kappa T_{,i})_{,i} + \frac{\overline{D}}{Ra}\tau_{ij}u_{i,j} + \rho_r H, \tag{3}$$

where ρ_r, u, p, τ, T, T_r, T_s, γ, κ and H are the vertical density profile, velocity, dynamic pressure, deviatoric stresses, temperature, reference temperature, surface temperature, mantle compressibility, thermal diffusivity and heat production rate, respectively; i and j are the spatial indices and 2 means in the vertical direction; δ is the Kronecker delta function; $\dot{T}$ is the local time rate of change of temperature; Ra is the thermal Rayleigh number and $\overline{D}$ is the dissipation number.

[12] The Rayleigh number Ra, dissipation number $\overline{D}$ and mantle compressibility γ are defined as

$$Ra = \frac{\rho_0 \alpha_0 g_0 \Delta T d^3}{\kappa_0 \eta_0}, \tag{4}$$

$$\overline{D} = \frac{\alpha_0 g_0 d}{C_{p\_0}}, \tag{5}$$

$$\gamma = \frac{\overline{D}}{\Gamma_0}, \tag{6}$$

where ρ_0, α_0, g_0, ΔT, d, κ_0, η_0, C_{p_0} and Γ_0 are dimensional numbers of surface density, surface thermal expansivity, surface gravitational acceleration, temperature contrast between surface and the core-mantle boundary (CMB), mantle thickness, surface thermal diffusivity, reference viscosity, surface specific heat at constant pressure and surface Grueneisen parameter, respectively. Equations (1)–(3) are nondimensionalized with the following scalings: $x_i = x_i'/d$, $t = t'\kappa_0/d^2$, $T = (T' - T_s')/\Delta T$, $p = p'd^2/(\eta_0\kappa_0)$, $H = d^2 H_0/(\kappa_0 \Delta T C_{p\_0})$, where x_i, t and H_0 is the coordinate in direction i, time, and internal heat production rate, respectively. Symbols with primes are corresponding dimensional values. Details of the derivations of these equations can be found in the study by *Leng and Zhong* [2008]. We use the Adams-Williamson equation for density distribution along vertical direction and obtain [*Leng and Zhong*, 2008]

$$\rho_r(z) = \exp[(1 - z)\gamma], \tag{7}$$

where z is the nondimensional vertical coordinate.

2.1.2. Thermochemical Mantle Convection

[13] For compressible thermochemical convection, equation (2) becomes

$$-p_{,j}\delta_{ij} + \tau_{ij,j} + [\rho_r Ra(T - T_r) - p\gamma - RbC]\delta_{i2} = 0, \tag{8}$$

where Rb is the chemical Rayleigh number which is defined as $Rb = \Delta\rho g_0 d_0^3/(\kappa_0\eta_0)$ with $\Delta\rho$ as the density difference due to chemical composition variation; C is the chemical composition. When using a tracer method, the movement of particles follows the conservation equation of compositions

$$\dot{C} + u_i C_{,i} = 0, \tag{9}$$

where $\dot{C}$ is the local time rate of change of chemical composition.

2.1.3. The Solvers of the Equations

[14] Based on the finite element method, we use a stabilized Q1-Q1 scheme to represent velocity and pressure field [*Dohrmann and Bochev*, 2004; *Elman et al.*, 2005; *Burstedde et al.*, 2009]. This scheme is effective for removing the checkerboard pattern of pressure field which is normally observed with a Q1-P0 scheme [*Hughes*, 2000].

[15] The matrix form of equations (1) and (2) can be expressed as

$$\begin{bmatrix} K & G^T + W \\ G + X & -D \end{bmatrix} \begin{bmatrix} V \\ P \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix}, \tag{10}$$

where K, G^T and G are the stiffness matrix, the discrete gradient operator and the discrete divergence operator, respectively. V, P and F are the velocity vector, pressure vector and force vector, respectively. D is the stabilization matrix for Q1-Q1 scheme [*Elman et al.*, 2005, p. 243]. W shows the effects of dynamic pressure on the buoyancy force, and X shows the effects of mantle compressibility [*Leng and Zhong*, 2008].

[16] We solve the matrix equation (10) for the compressible Stokes flow problem with the modified Uzawa method [*Leng and Zhong*, 2008], and solve the energy equation (3) with the streamline upwind Petrov-Galerkin method [*Brooks and Hughes*, 1982] that has also been used in mantle convection code Conman and Citcom [*King et al.*, 1990; *Moresi et al.*, 1996]. The chemical advection equation (9) is solved with a predictor-corrector and second-order Runge-Kutta scheme [*Zhong and Hager*, 2003; *McNamara and Zhong*, 2004].

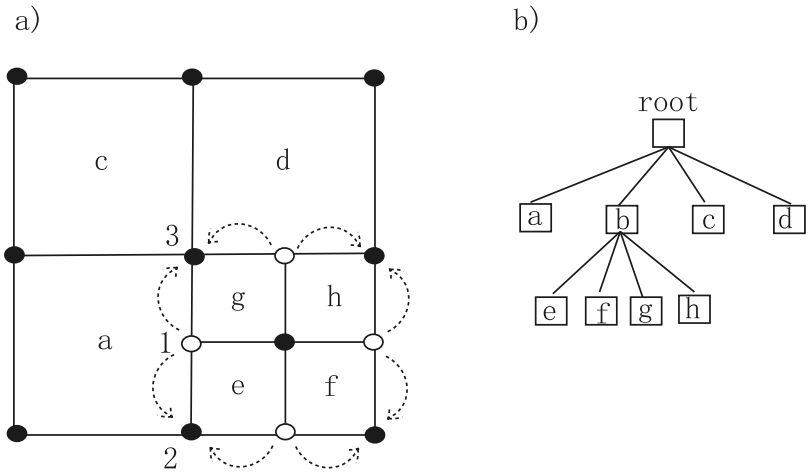

Figure 1. The (a) mesh distribution and (b) corresponding quadtree data structure. In Figure 1a, the open circles show the hanging nodes generated due to the mesh refinement. The arrows show that the values at the hanging nodes are constrained by their master nodes.

2.2. Adaptive Mesh Refinement Technique

2.2.1. The Quadtree Data Structure and the Morton Code

[17] To solve the governing equations with the finite element method, first of all the computational domain needs to be discretized, forming a mesh that contains elements and nodes. To implement the adaptive mesh refinement techniques, a good approach for decomposing the domain is based on the quadtree data structure in 2-D, or octree data structure in 3-D [*Tu et al.*, 2005; *Burstedde et al.*, 2008; *Stadler et al.*, 2010]. Figure 1 shows the representation of a mesh distribution and the corresponding quadtree data structure that we used from *Tu et al.* [2005]. Basically, each element in the domain is represented by a leaf in the main quadtree. When an element is refined, several child elements (four for 2-D and eight for 3-D) are added to the tree structure in a lower layer. In Figure 1, the element b is subdivided into four elements (e, f, g, h) that are inserted into the quadtree in a lower layer to replace the element b. Only the volumes represented by the leaves in the quadtree (a, e, f, g, h, c, d) will be used as elements in the finite element computation. The child elements can also be simply removed from the quadtree structure when high resolution is no longer necessary. Then the refined elements (e, f, g, h) are coarsened back to the element b. Based on the quadtree data structure, the AMR can dynamically adjust mesh distribution and put high resolution in regions where fine structures need to be resolved.

[18] Notice that when an element is refined, new nodes are generated at the center of the element and possibly at the midpoint of the edges of the element (Figure 1a). For the new node at the element center, the velocity, temperature, and the local time rate of change of temperature are interpolated as the averaged values from the four nodes of the refined element. For the new nodes at the midpoint of each edge, the velocity, temperature, and the local time rate of change of temperature are interpolated as the averaged values from the two endpoints of that edge. When an element is coarsened, we simply take the values on the nodes that remain after coarsening.

[19] In order to effectively search and map between the mesh distribution and the quadtree data structure, it is very useful to construct another linear sequential structure for the quadtree with a Morton code technique [*Morton*, 1966; *Tu et al.*, 2005]. Each element in the mesh distribution is uniquely assigned a Morton code which is constructed from the coordinates of the element. The location of the elements in the quadtree then can be easily retrieved from the Morton code (see *Tu et al.* [2005] for the details). This feature facilitates searching and locating elements in the quadtree. Since the Morton code constructed for the whole domain is a linear sequence, it is also convenient to partition the quadtree data structure into different segments for parallel computation [*Tu et al.*, 2006; *Burstedde et al.*, 2008; *Stadler et al.*, 2010].

2.2.2. The Hanging Nodes and the Constrained Matrix Equation

[20] When an element is refined, it is required that two adjacent elements which share an edge should

not differ in edge size by a factor of 2. This constraint is usually called the balance condition or the 2-to-1 constraint [*Bank et al.*, 1983; *Tu et al.*, 2005, 2006]. There are several practical and theoretical reasons for imposing this constraint. A detailed argument is referred to *Bank et al.* [1983]. Following this constraint, when the element b in Figure 1 is refined, four nodes hanging at the midpoint of the edges are generated and marked as the hanging nodes (open circles). For conforming finite element methods, the values (velocities, temperature, etc.) at each hanging node have to be constrained as the average values of its master nodes. The master nodes for a hanging node are defined as the endpoints of the edge where the hanging node lies. For example, the velocity at the hanging node 1 in Figure 1a is constrained as the average velocity of its master nodes 2 and 3.

[21] With the constraints for the hanging nodes, the matrix equation (10) needs to be modified to incorporate the constraints. A typical way to solve matrix equation with constraints in finite element analysis is developed by *Abel and Shephard* [1979] and *Shephard* [1984]. They partitioned the matrix equations into submatrices of constrained and unconstrained degrees of freedom. The constrained degrees of freedom are then removed by matrix transformation. For example, suppose we want to solve the matrix equation

$$KV = Y, \tag{11}$$

where K, V, and Y are the stiffness matrix, velocity vector and load vector. We first partition equation (11) to equation of submatrices as

$$\begin{bmatrix} K_{ll} & K_{lm} & K_{ln} \\ K_{ml} & K_{mm} & K_{mn} \\ K_{nl} & K_{nm} & K_{nn} \end{bmatrix} \begin{bmatrix} V_l \\ V_m \\ V_n \end{bmatrix} = \begin{bmatrix} Y_l \\ Y_m \\ Y_n \end{bmatrix}, \tag{12}$$

where V_l, V_m and V_n represent velocities not related to hanging nodes and master nodes, velocities of master nodes and velocities of hanging nodes. Since velocities of hanging nodes are constrained by their master nodes, we have the constraining equation

$$V_n = A_{nm} V_m, \tag{13}$$

where A_{nm} is a constraining matrix with entries of 0.5 and 0.0. By substituting equation (13) into equation (12), we can remove V_n from the matrix equation (see *Abel and Shephard* [1979] for the details of matrix transformation) and obtain new matrix equation as

$$\begin{bmatrix} K_{ll} & K_{lm} + K_{ln} A_{nm} \\ K_{ml} + A_{nm}^T K_{nl} & K_{mm} + K_{mn} A_{nm} + A_{nm}^T K_{nm} + A_{nm}^T K_{nn} A_{nm} \end{bmatrix} \cdot \begin{bmatrix} V_l \\ V_m \end{bmatrix} = \begin{bmatrix} Y_l \\ Y_m + A_{nm}^T Y_n \end{bmatrix}, \tag{14}$$

where A_{nm}^T is the transpose of A_{nm}. *Shephard* [1984] further improved this method by performing the matrix transformation on each individual term of the stiffness matrix through algebraic operations. We use *Shephard*'s [1984] method due to its advantage that the modified stiffness matrix can be constructed directly from the element stiffness matrix.

2.3. The Improved Tracer Method With Adaptive Mesh Refinement

[22] Using particles as tracers to simulate the different chemical compositions in mantle convection is a common method for thermochemical convection studies [*van Keken et al.*, 1997; *Tackley and King*, 2003; *McNamara and Zhong*, 2004]. Here we employ an improved tracer method in our AMR code, which is similar to the previous tracer ratio method [e.g., *Tackley and King*, 2003] but with modifications for incorporating mesh refinement and coarsening.

[23] In our tracer method, two different kinds of particles are introduced to represent two different compositions with $C = 0$ and $C = 1$. Initially, the particles are uniformly distributed throughout the computational domain and the compositions of the particles are assigned from the initial chemical composition field. The composition of an element is determined by the ratio of these two different particles in the element. For example, if there are $n1$ particles with $C = 0$ and $n2$ particles with $C = 1$ in an element, the composition of the element is determined as $n2/(n1 + n2)$. With the advection of the particles, the evolution of the chemical composition field can be simulated.

[24] We solve the particle distribution problem with dynamical adjustment of the number of particles in each element. Initially, we put 25 particles uniformly distributed in each element. This initial particle number is not critical since the number of particles in each element will be frequently modified with mesh adjustment. We then set the maximum number of particles, N_{max}, and the minimum number of particles, N_{min}, that each element may contain. Every time when the meshes are adjusted

with the AMR techniques, the number of particles in each element, N_e, is checked. If N_e is smaller than $N_{\min}$ and the element composition is greater than 0.99 or smaller than 0.01, we add a number of P_a particles into the elements. The positions of these new particles are randomly distributed in the element. The compositions of these particles are determined by the composition of the element (i.e., if the element composition is larger than 0.99, we add particles with $C = 1$; if the element composition is smaller than 0.01, we add particles with $C = 0$). If N_e is larger than $N_{\max}$, we randomly remove particles in the element by a percentage of P_r. In order to decide which particle in the element should be removed, we generate a random number between 0 and 1 for each particle. If the number is smaller than P_r, the particle is removed. Otherwise, the particle is preserved. Our experience suggests that the following numbers are effective for simulating chemical compositions in our 2-D AMR code: $N_{\min} = 4$, $N_{\max} = 30$, $P_a = 4$ and $P_r = 50\%$. Small variations to these numbers cause negligible influences to our results. We therefore keep using these numbers in our AMR code. These numbers may need to be adjusted for more complicated or realistic mantle flow problems (e.g., mantle flow with very high Rayleigh number). The principle is to maintain certain amount of particles in each element to preserve mass conservation.

2.4. The Rules for Mesh Refinement and Coarsening

[25] We define two parameters, M_l and M_h, for describing the extent of mesh refinement in our 2-D AMR model. M_l and M_h are the lowest and the highest level for the mesh refinement in each direction, which means that the number of elements are restricted between 2^{M_l} and 2^{M_h} in both horizontal direction and vertical direction. For example, if M_l and M_h are chosen as 3 and 5, the possible number of elements in our model is then between $2^3 \times 2^3$ and $2^5 \times 2^5$.

[26] One important problem for the AMR technique is to construct a good error indicator which guides the refinement and coarsening of elements [e.g., *Babuvška and Rheinboldt*, 1978; *Verfurth*, 1996; *Ainsworth and Oden*, 2000]. The design and implementation of a perfect error indicator can be quite complicated and is not the focus of this study. Here we use several rules as discussed below. These rules are relatively simple and straightforward. However, we think they are suitable for this study and good enough to demonstrate our points.

[27] For the B_{thre} rule, an element is refined to M_h level if it is at the top or bottom boundaries. This rule is introduced to ensure that the top and bottom thermal boundary layers are well resolved in thermochemical convection.

[28] For the T_{thre} rule, an element is refined if the horizontal or vertical temperature gradient is greater than 0.05. For mesh coarsening process, if the horizontal and vertical temperature gradients are both smaller than 0.025 for all the subelements (e.g., e, f, g, and h in Figure 1a), these subelements are combined into one element through mesh coarsening processes.

[29] For the C_{thre} rule, if the chemical composition of an element, C, is between 0.01 and 0.99, this element is considered lying at the chemical boundaries and is refined. For mesh coarsening, if C is smaller than 0.01 or larger than 0.99 for all the subelements (e.g., e, f, g, and h in Figure 1a), these subelements are combined into one element through mesh coarsening processes.

[30] When two or more rules are used together, we refine an element if any rule is satisfied for the element. Notice that when an element is refined, we also refine a number of N_r elements which are adjacent to the refined element in all four directions (top, bottom, left and right). This treatment ensures that as the simulation evolves, solution structures that require high resolution do not move into coarser regions of the mesh. From our current testing results, we choose $N_r = 5$, although varying N_r between 3 and 7 has negligible effects on our results. A more effective way is to only refine the neighbor elements in the advection direction instead of in all four directions. This is a worthwhile point to be explored in future studies.

3. Benchmark Results of the Adaptive Mesh Refinement Code

3.1. Stokes Flow

[31] To verify our 2-D AMR code, we first solve the Stokes flow problem (i.e., equation (10)) with our code and compare the results with the analytic solutions from the propagator matrix method [*Hager and O'Connell*, 1979; *Leng and Zhong*, 2008]. We start from an isoviscous case SC1 in a 1×1 box. The compressibility γ is set to be zero for this case. The kinematic boundary conditions are free slip for all the boundaries. Given the Rayleigh number $Ra = 1.0$, the buoyancy force is represented by a delta

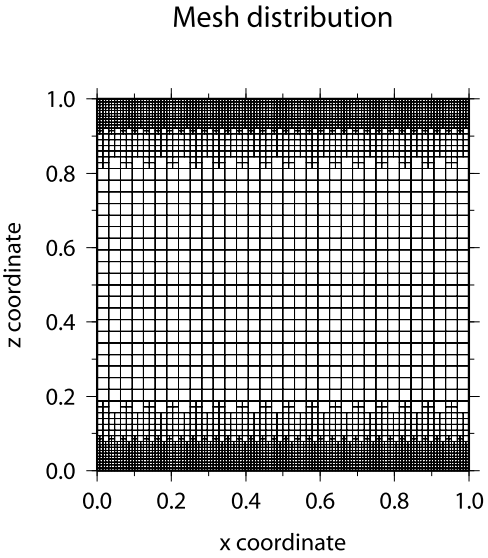

Figure 2. The refined mesh distribution for case SC1.

function of temperature perturbation at the middepth of the box,

$$T(x,z) = \delta(z-0.5)\cos(\pi x), \tag{15}$$

where x and z are the nondimensional coordinates in horizontal and vertical direction; δ is the Kronecker delta function. With the propagator matrix method, the analytic solutions of the horizontal velocity (V_s) and vertical normal stress (S_{rr}) at the surface are obtained and shown in Table 1. For case SC1, the refinement parameters M_l and M_h are set to be 5 and 7, respectively. We simply use B_{thre} to refine the boundary elements in the mesh distribution as shown in Figure 2. The numerical results for case SC1 are computed and also shown in Table 1. Notice that these numerical results represent the maximum horizontal velocity and the maximum vertical normal stress at the surface. It can be observed that the differences between the analytic solutions and the numerical results are very small. The relative difference for the surface horizontal velocity ε_V is 0.2% and the relative difference for the surface normal stress ε_S is 0.1%. Case SC2 is the same as case SC1 except that mantle compressibility $\gamma = 0.5$ is introduced. From equation (7), the density increases by a factor of 1.65 from the surface to the bottom. Similar to case SC1, ε_V and ε_S are both very small, 0.1% for case SC2 (Table 1).

[32] While cases SC1 and SC2 are both isoviscous with viscosity equal to 1.0 everywhere, case SV1 and SV2 use a layered viscosity structure that still allows analytic solutions [*Leng and Zhong*, 2008]. Case SV1 is the same as case SC1, except that case SV1 has two layers of different viscosities: viscosity is 1000 and 1 above and below $z = 0.84375$, respectively. Since this is a difficult case to be computed, we use this case to compare our AMR results with results from uniform meshes. We fix $M_l = 4$ and vary M_h from 4 to 7 to compute cases SV1a, SV1b, SV1c and SV1d as shown in Table 1. Specifically, we compute the relative error for the horizontal velocity u_1 across the surface as $\|u_1 - u_a\|/\|u_a\|$, where $\|$ denotes the L_2 norm and u_a is the analytic solution. Figure 3 shows the relative error versus the number of elements that we used with AMR. In Figure 3, we also plot the results computed for case SV1 but with uniform meshes 16 × 16, 32 × 32, 64 × 64, and 128 × 128. It can be observed that our AMR code can obtain similar results with less elements than the uniform meshes (Figure 3), thus is efficient in memory requirements. Case SV2 is the same as case SV1, except that the mantle compressibility $\gamma = 0.5$ is included and M_l and M_h are set to be 5 and 7. Table 1 shows that the numerical results and the analytic results also agree very well with each other for case SV2.

Table 1. Benchmark of the AMR Code for Stokes Flow[a]

Cases	γ	$V_s(ana)$	V_s	$\varepsilon_V(\%)$	$S_{rr}(ana)$	S_{rr}	$\varepsilon_S(\%)$
SC1	0.0	0.04569	0.04559	0.2	0.4863	0.4858	0.1
SC2	0.5	0.06173	0.06165	0.1	0.6096	0.6091	0.1
SV1a	0.0	9.714e-5	1.053e-4	8.4	0.5799	0.5597	3.5
SV1b	0.0	9.714e-5	1.001e-4	3.0	0.5799	0.5699	1.7
SV1c	0.0	9.714e-5	9.759e-5	0.5	0.5799	0.5721	1.3
SV1d	0.0	9.714e-5	9.757e-5	0.4	0.5799	0.5726	1.3
SV2	0.5	1.201e-4	1.206e-4	0.4	0.6753	0.6734	0.3

[a]The γ, V_s, and S_{rr} are the mantle compressibility, horizontal velocity at the surface, and radial stress at the surface, respectively. Values with "(*ana*)" represent the results from analytic solutions. The $\varepsilon_V(\%)$ and $\varepsilon_S(\%)$ represent the relative errors between the analytic solutions and the numerical results for V_s and S_{rr}, respectively. Cases SC1 and SC2 have constant viscosity. Cases SV1 and SV2 have layered viscosity. Notice that for all these cases M_l and M_h are set to be 5 and 7, except for cases SV1a–SV1d, where $M_l = 4$ is fixed and M_h increases from 4 to 7.

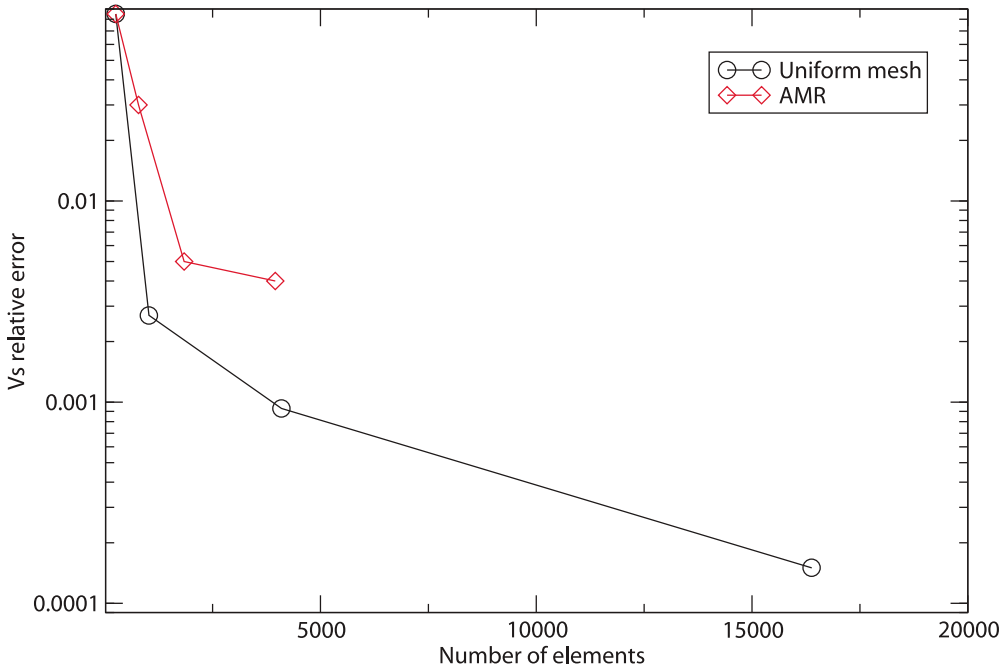

Figure 3. The relative error of horizontal velocity versus number of elements for case SV1 using AMR and uniform meshes. For the AMR, we fix $M_l = 4$ and varies M_h from 4 to 7; for the uniform meshes, we use 16 × 16, 32 × 32, 64 × 64, and 128 × 128 elements, respectively.

3.2. Thermal Convection

[33] By solving equation (10) with the energy conservation equation (3), our code can be used to study thermal convection problems. We first compute a thermal convection case TC1 with Ra = 1.0e4 and $\gamma = \overline{D} = H = 0$. This case is the same as case 1a in the benchmark paper of incompressible mantle convection by *Blankenbach et al.* [1989]. The box dimension is 1 × 1 and the kinematic boundary conditions are free slip for all the boundaries. The top and bottom boundaries are isothermal and the nondimensional temperatures are fixed as 0.0 and 1.0 at the top and bottom boundaries, respectively. The sidewall boundaries are thermally insulated. We use an initial temperature profile which linearly increases from 0.0 at the top to 1.0 at the bottom and a small perturbation to start the convection. For the mesh refinement, we fix $M_l = 4$ and varies M_h from 4 to 7 to compute cases TC1a, TC1b, TC1c and TC1d as shown in Table 2. We use both B_{thre} and T_{thre} to refine the meshes.

[34] We run cases TC1a–TC1d to steady state. Figure 4a shows the mesh distribution and the nondimensional temperature field after case TC1d reaches the steady state. It can be observed that elements with large temperature gradients are refined. The Nusselt number Nu and root-mean-

Table 2. Benchmark of the AMR Code for Thermal Convection[a]

Cases	γ	Nu	Nu_b	ε_N(%)	V_{rms}	V_b_{rms}	ε_R(%)
TC1a	0.0	4.8935	4.8844	0.2	43.04	42.86	0.4
TC1b	0.0	4.8904	4.8844	0.1	42.97	42.86	0.3
TC1c	0.0	4.8818	4.8844	0.05	42.94	42.86	0.2
TC1d	0.0	4.8855	4.8844	0.02	42.99	42.86	0.3
TC2	0.0	10.077	10.066	0.1	485.8	480.4	1.1
TC3	0.5	5.4533	5.4240	0.5	241.7	238.6	1.3

[a]The γ, Nu, and V_{rms} are the mantle compressibility, Nusselt number, and root-mean-square velocity for the whole box, respectively. Nu_b and V_b_{rms} are the benchmark results published before for the Nusselt number and root-mean-square velocity. The ε_N(%) and ε_R(%) represent the relative errors between our numerical results and previously published benchmark results for Nu and V_{rms}, respectively We obtain the benchmark results for the cases TC1 and TC2 from *Blankenbach et al.* [1989] and for the case TC3 from *King et al.* [2010]. Notice that for cases TC1a–TC1d, $M_l = 4$ is fixed and M_h increases from 4 to 7, while for cases TC2 and TC3, M_l and M_h are set to be 5 and 7, respectively.

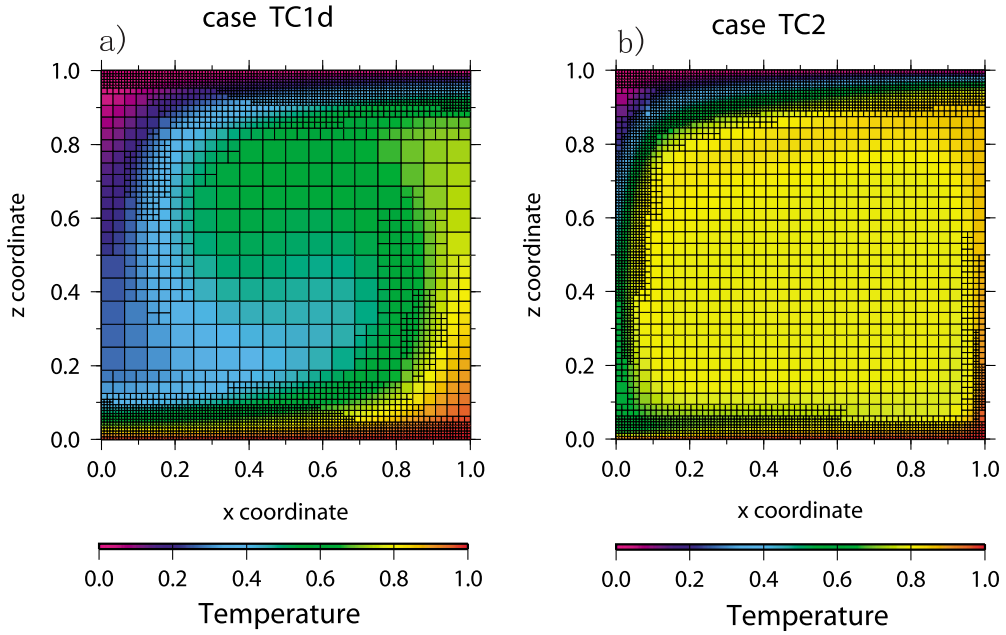

Figure 4. The nondimensional temperature field and mesh distribution (a) for case TC1d with constant viscosity and (b) for case TC2 with temperature-dependent viscosity at steady state.

square (RMS) velocity V_{rms} are computed and compared with the benchmark results published by *Blankenbach et al.* [1989]. The relative errors for Nu and V_{rms} are both very small for these cases (Table 2). To compare the results from AMR and uniform meshes, we compute three cases with uniform resolution and plot their surface heat flux results in Figure 5. These three cases are identical to case TC1, except that they are using uniform resolution as 32 × 32, 64 × 64, and 128 × 128, respectively. The surface heat flux for case TC1 is also computed with $M_l = 5$ and $M_h = 7$ and shown in Figure 5. The comparison of surface heat flux shows that case TC1 with AMR techniques achieves the same accuracy for the surface heat flux as the case with uniform resolution 128 × 128 (Figure 5), although case TC1 uses only a quarter of elements (i.e., 3952 elements for the case TC1).

[35] To show the time efficiency of our AMR code, we plot the relative error of Nu for case TC1 versus the CPU time (Figure 6). In Figure 6, the black line shows the results from uniform mesh 64 × 64, and the red line shows the results from AMR with $M_l = 5$ and $M_h = 7$. It can be observed that it takes less time for our AMR code to achieve equal or better accuracy compared with uniform meshes. *Davies et al.* [2007] obtained similar results as shown in their Figure 7, but their curves are smoother than ours. We consider this difference is possibly caused by the fact that they used direct solver and unstructured meshes while we use iterative solver and structured meshes. We compute the time required for mesh adjustment for the AMR case shown in Figure 6 and compare with the total CPU time. For the total CPU time of 10000 s, the computation time for the mesh refinement and coarsening only accounts for ~0.03%.

[36] Case TC2 is the same as case 2a in the study by *Blankenbach et al.* [1989], differing from case TC1 in that it has a temperature-dependent

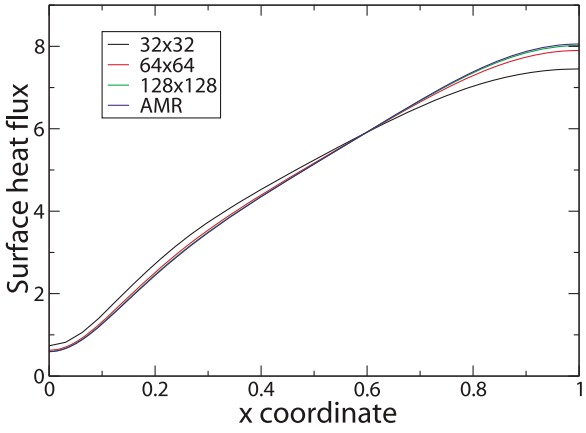

Figure 5. The surface heat flux for case TC1 after it reaches a steady state computed using AMR and three uniform meshes. Those cases with uniform meshes are identical to case TC1 except for the different resolution.

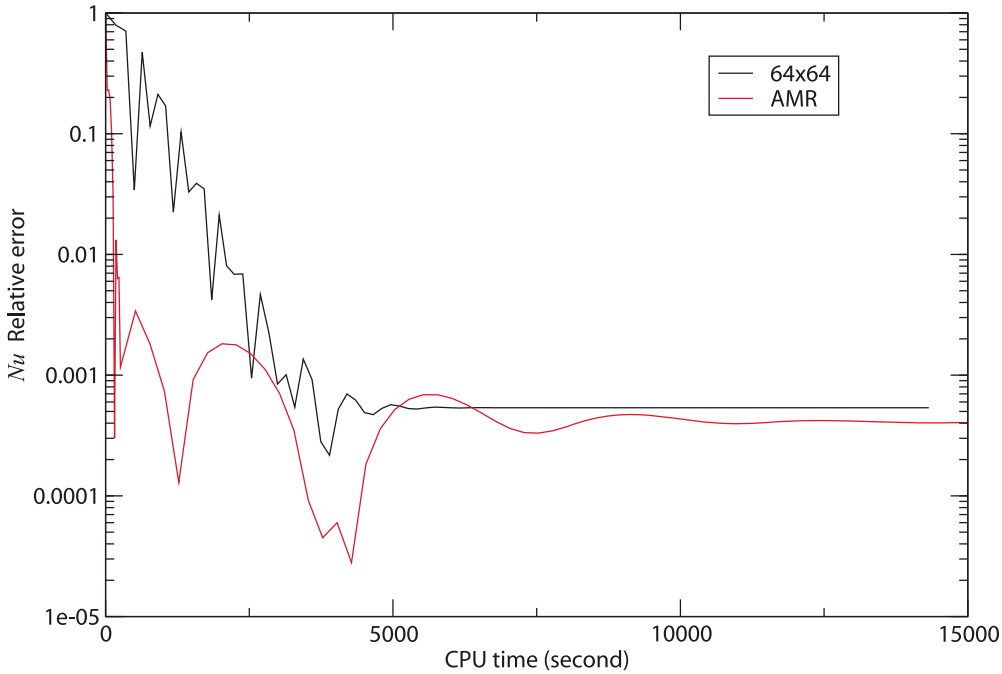

Figure 6. The relative error of *Nu* versus CPU time from case TC1 using AMR and uniform meshes. For the AMR, $M_l = 5$ and $M_h = 7$; for the uniform meshes, we use 64 × 64 elements.

viscosity $\eta = \exp(-ET)$. Here, η is the viscosity and $E = \ln(1000)$ is the nondimensional activation energy. We use $M_l = 5$ and $M_h = 7$ for case TC2. Figure 4b shows the mesh distribution and the nondimensional temperature field after case TC2 reaches steady state. The Nusselt number and RMS velocity for case TC2 also agree well with the results by *Blankenbach et al.* [1989], i.e., 0.1% and 1.1% difference for Nusselt number and RMS velocity, respectively (Table 2). Case TC3 is the same as case TC2, except that the mantle compressibility γ and mantle dissipation $\overline{D}$ number are both 0.5 and the nondimensional surface temperature is, $T_s = 273/3000 = 0.091$. This case is the same case as shown in section 5.2 in the study by *King et al.* [2010] with $\overline{D} = 0.5$. The comparison between our results and the benchmark results in the study by *King et al.* [2010] shows very small difference, i.e., 0.5% and 1.3% difference for Nusselt number and RMS velocity, respectively (Table 2).

3.3. The Single Vortex Flow

[37] We test our modified tracer method with a single vortex flow problem which is the same as the one used by *Hieber and Koumoutsakos* [2005, section 5.2]. Initially, a chemical circle with a radius of 0.15 is put at (0.5, 0.75) in a 1 × 1 box and filled with particles of $C = 1$ (Figure 7a). The outside area is filled with particles of $C = 0$. The velocity field in the box is prescribed as a single vortex flow:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = 2 \begin{bmatrix} -\sin^2(\pi x)\sin(\pi z)\cos(\pi z) \\ \sin^2(\pi z)\sin(\pi x)\cos(\pi x) \end{bmatrix}. \quad (16)$$

We use the chemical composition criterion C_{thre} to refine the meshes and choose $M_l = 4$ and $M_h = 8$, respectively.

[38] At $t = 3.0$, the circle is stretched to be a very long and thin filament (Figure 7b). The corresponding mesh distribution is shown in Figure 7c. It can be seen that our improved tracer method together with the AMR technique is fully capable of resolving the stretched structure with sharp boundaries.

[39] This pure advection problem can be used to test the mass conservation of the chemical field. If the velocity field in the equation (16) is multiplied by $\cos(\pi t/L)$, the circle will first be stretched to a long filament, then be restored to its original shape within a period of L. In this study, we choose $L = 8.0$. By comparing the area error between the original shape (i.e., the exact area of the circle is 0.07069) and the restored shape, the mass conservation of the chemical field can be tested. The area of the restored shape is numerically computed as the volume integral of chemical composition.

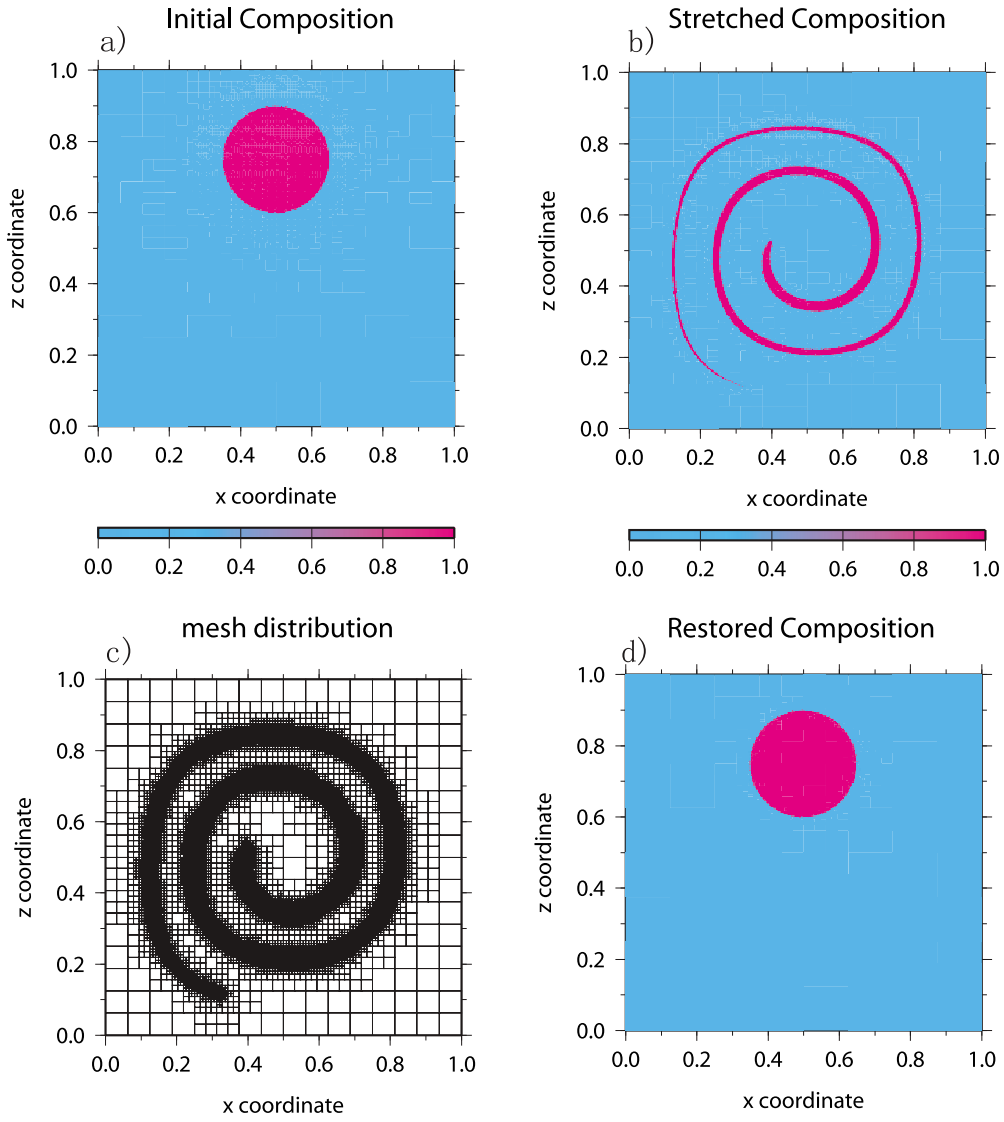

Figure 7. (a) The initial composition field for the single vortex flow case with $M_l = 4$ and $M_h = 8$. (b) The stretched composition field at $t = 3.0$ for the single vortex flow case. (c) The mesh distribution corresponding to Figure 7b. (d) The restored composition field for case PA04 at $t = 8.0$.

[40] We fix $M_l = 4$ and vary M_h from 4 to 8 (cases PA01–PA04 in Table 3). Figure 7d shows the well restored shape for case PA04 at $t = 8.0$. Notice that the neighbor element refinement parameter N_r is modified from 5 to 3 for these cases. This helps to reduce the number of particles we used. The quantitative results are shown in Table 3. The small errors show that the mass of the chemical field is well conserved, even for relatively low mesh resolution. As M_h increases, the required number of particles increases and the error slightly decreases (Table 3). Compared with the results by *Hieber and Koumoutsakos* [2005] and the references therein, our errors and required particles are similar to those in the hybrid particle level set method [*Enright et al.*, 2002]. We also fix $M_l = 4$ and $M_h = 6$ and vary N_r between 1 and 5 to observe its effects on our results (cases PA05–PA06 in Table 3). It can be noted that a larger N_r reduces the error but introduces more particles (Table 3).

3.4. Rayleigh-Taylor Instability

[41] We also verify the modified tracer method with a Rayleigh-Taylor problem case RT1. This problem is the same as the Rayleigh-Taylor problem tested in the study by *van Keken et al.* [1997]. We have a

Table 3. Results for the Single Vortex Flow[a]

Cases	M_l	M_h	N_r	Error (%)	Particles(t = 8)
PA01	4	5	3	1.2	6232
PA02	4	6	3	0.7	10726
PA03	4	7	3	0.06	19414
PA04	4	8	3	0.04	37065
PA05	4	6	1	0.9	8827
PA06	4	6	5	0.1	13551

[a]Here "Error" is the relative error of the area of the compositional field at $t = 8$ compared to the analytic result of 0.07069. The numbers listed in the "Particles" column indicate the number of particles within the domain at $t = 8$ for each case.

box of size 0.9142 × 1.0 filled with isoviscous fluid with viscosity equal to 1.0. Initially, the box includes two layers of fluids with different densities. The bottom layer is buoyant with a thickness of 0.2 and the particles are assigned with $C = 0$. The particles in the top layer are assigned with $C = 1$. The compositional interface is perturbed initially with $w = 0.02\cos(\pi x)$. We do not consider thermal and compressible effects, thus $Ra = \gamma = \overline{D} = H = 0$. The chemical Rayleigh number is $Rb = 1.0$. The top and bottom boundaries are nonslip and the lateral boundaries are free slip. For mesh refinement, we

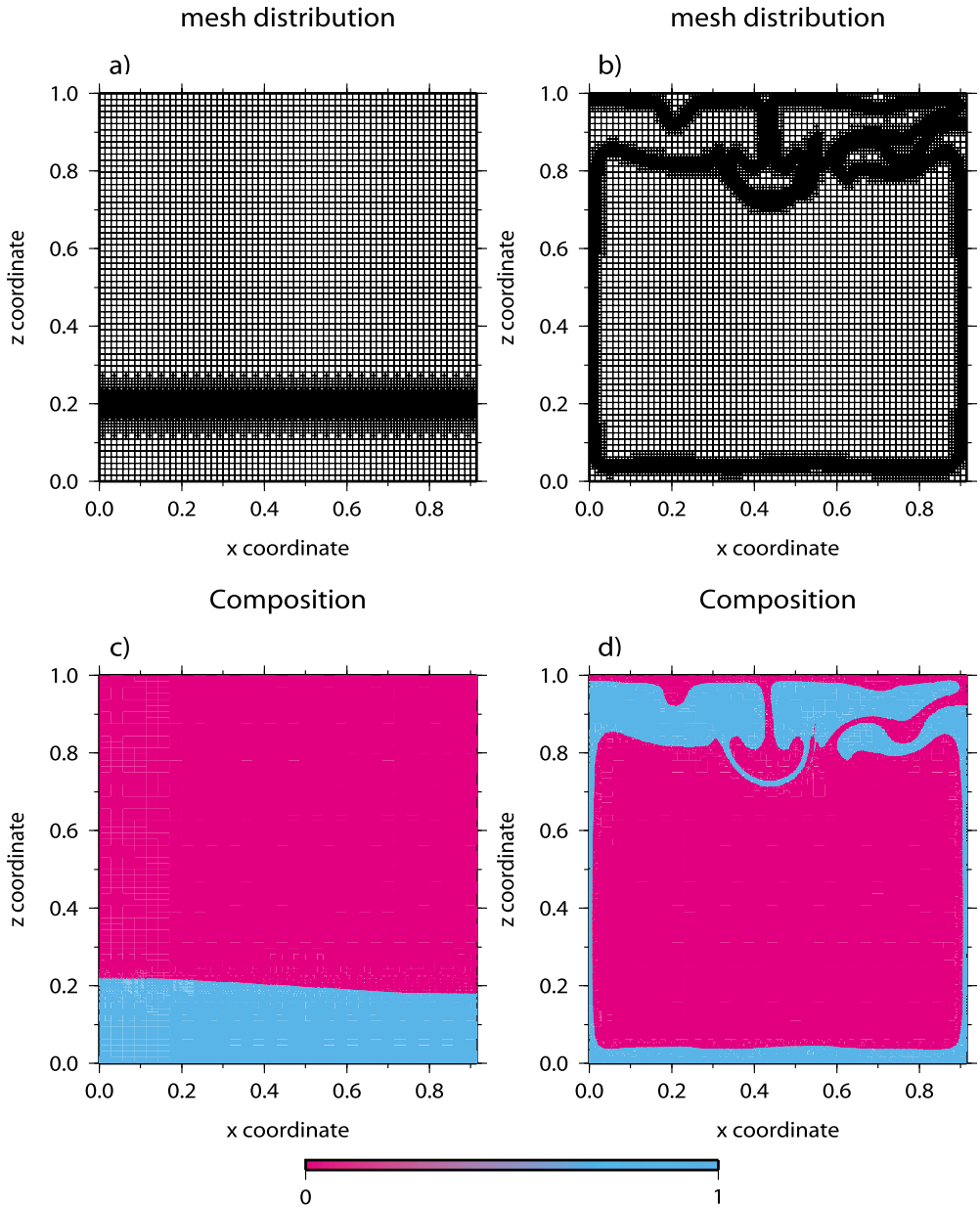

Figure 8. The mesh distribution and chemical composition for case RT1 at two different times: (a and c) $t = 0$ and (b and d) $t = 1500$.

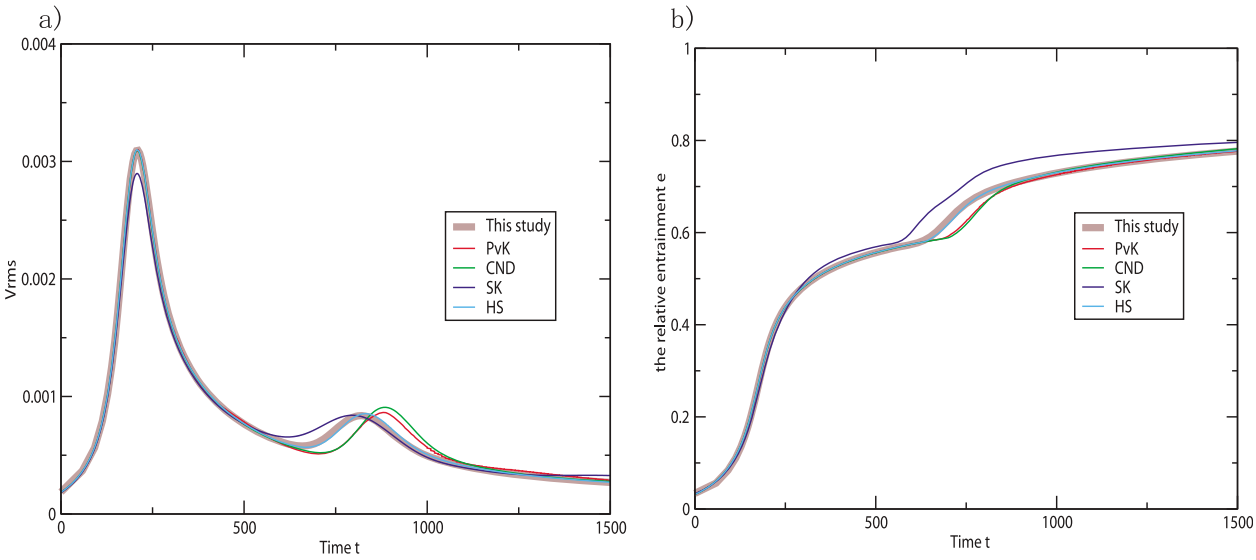

Figure 9. (a) The root-mean-square velocity V_{rms} and (b) the relative entrainment of the buoyant material, with time for case RT1. The corresponding benchmark results from *van Keken et al.* [1997] are also plotted. PvK, CND, SK, and HS are the representations of different groups that are indicated in the study by *van Keken et al.* [1997].

choose $M_l = 6$ and $M_h = 9$, respectively. We use the chemical composition criterion C_{thre} to refine and coarsen the mesh.

[42] Figure 8 shows the mesh distribution and chemical composition field for the case RT1 at two different times, $t = 0$ and $t = 1500$. Figures 8a and 8b show that the chemical boundaries are significantly refined and that the mesh refinement evolves with time to track the chemical boundaries. Visual comparison between Figure 8d here and Figure 2 by *van Keken et al.* [1997] shows that our code successfully reproduces the benchmark results by *van Keken et al.* [1997]. Figure 9 shows time dependence of root-mean-square velocity V_{rms} and the relative entrainment, e, of the buoyant material for the case RT1. The relative entrainment, e, is computed as the volume integral of the buoyant chemical composition above $z = 0.2$, and is normalized by the original total volume of the buoyant material. The benchmark results from different codes in the study by *van Keken et al.* [1997] are also shown in Figure 9. It can be observed that our results are very similar to the results from other groups. The results from different groups have negligible differences for the generation of the first instability except for the results from SK, but all have some small difference at $t \sim 750$ when the second instability develops.

[43] To show the time efficiency of our AMR code for the Rayleigh-Taylor problem, we compute three cases with uniform resolution as 64 × 64, 128 × 128, and 256 × 256, respectively. The CPU time for these three cases to reach the peak of the first instability (i.e., at t ~ 210, see Figure 9a) is 502, 7154 and 85738 s, respectively. For the AMR case discussed above, the corresponding CPU time is 22423 s, which is much less than the case with uniform resolution of 256 × 256, although the highest local resolution of the AMR case is $M_h = 9$.

4. Computing the Entrainment Rate in the Thermochemical Mantle Convection

[44] In this section, we demonstrate that our AMR code may be employed to study entrainment problems in thermochemical convection which need high resolution to resolve chemical boundaries. Although previous numerical models have been able to use specific mesh setup to quantify the entrainment rate of a dense layer by mantle plumes [*Zhong and Hager*, 2003; *Huang*, 2008], quantifying the entrainment rate for a general thermochemical convection (i.e., the positions of upwelling plumes are not preset) is still a challenging problem.

[45] We start from a case ET1 which is similar to the thermal convection case TC1 discussed in the section 3.2, except that the Rayleigh number Ra is increased from 10^4 to 10^5 and that the top and bottom boundaries are nonslip. Case ET1 uses a constant viscosity of 1.0. We add a dense layer at

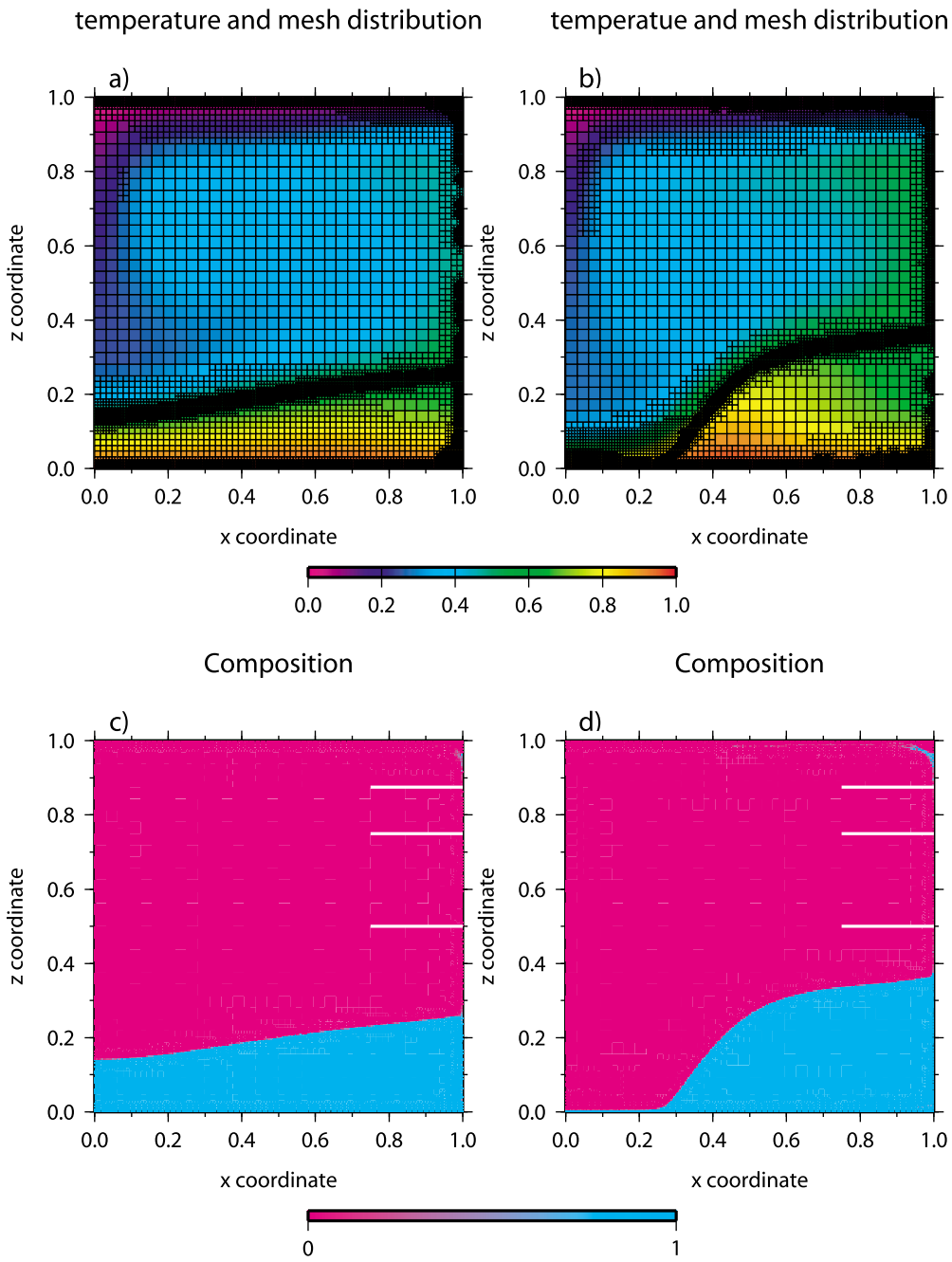

Figure 10. The mesh distribution, nondimensional temperature, and chemical composition after the thermochemical convection cases reach a quasi steady state; (a and c) for case ET1 with the buoyancy number $B = 1.2$ and (b and d) for the case with the buoyancy number $B = 0.6$. The white lines in Figures 10c and 10d are the lines over which the chemical flux is integrated.

the bottom of the box after running the case to a thermally steady state. The initial thickness of the dense layer is 0.2. The chemical buoyancy number B, which is defined as $B = Rb/Ra$, is 1.2 for the case ET1. For the mesh refinement, we use B_{thre}, T_{thre}, and C_{thre} to refine the meshes. We choose $M_l = 5$ and $M_h = 8$.

[46] After the dense layer is added, it starts to be gradually entrained by the upwelling plumes. Figures 10a and 10c show the mesh distribution, nondimensional temperature and chemical composition after case ET1 reaches a quasi steady state with a dense layer at the bottom. Since we keep adding and removing particles from the elements to

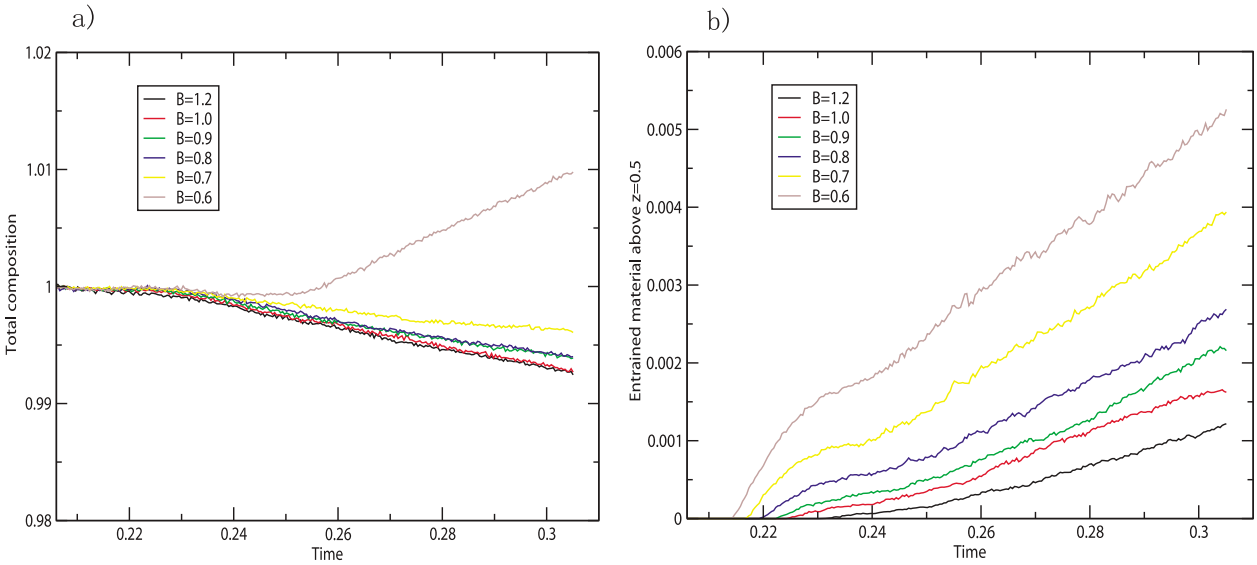

Figure 11. (a) The time dependence of normalized total composition and (b) the entrained dense material above $z = 0.5$, for the thermochemical convection cases with different chemical buoyancy number B.

adjust the particle distribution, it is useful to examine whether the total chemical composition is conserved during the particle adjustment. We compute the volume integral of the total dense composition over the whole box for the case ET1 and normalize it with the original total volume of the dense layer. Figure 11a shows the variation of the normalized total composition with time. It can be observed that the total composition varies less than 1% for the whole computational process, indicating that the conservation of chemical composition is well satisfied. Notice that we add the dense layer only after case ET1 reaches thermally steady state at $t = 0.206$, consequently the total composition is computed starting from time $t = 0.206$ instead of $t = 0.0$.

[47] We compute the entrainment rate of the dense material for case ET1 with two different methods. In the first method, we first determine the time dependence of the total volume of the entrained dense material above $z = 0.5$ (i.e., the integral of C for z greater than 0.5, Figure 11b), then the entrainment rate is computed from the rate of the entrained material with time (i.e., the slope of Figure 11b) after the entrainment reaches steady state [*Zhong and Hager*, 2003]. In Figure 11b, we consider that after time $t = 0.25$ the entrainment rate for case ET1 reaches the steady state. The time-averaged entrainment rate is computed thereafter. The second method is to directly compute the chemical flux of the dense material in the upwelling plume area above the dense layer. The chemical flux is represented as Cu_2, where C and u_2 are the chemical composition and the vertical velocity. The entrainment rate is then computed as the integrated chemical flux over a horizontal line at a proper height. In this study, we compute the entrainment rates with chemical flux over three different lines with $z = 0.5$, 0.75 and 0.875 (Figure 10c, three white lines). If numerical errors are small, the entrainment rates determined from these two methods should be consistent with each other. The results are shown in Figure 12. For case ET1 with $B = 1.2$, the red circle shows result from the first method and the three green squares show the integrated chemical flux over the three white lines. The entrainment rates measured by the two approaches are consistent with each other. Moreover, the results obtained from the integrated chemical flux over the three white lines produce similar results. This indicates that our methods for quantifying the entrainment rate are robust. However, the time-averaged entrainment rate show rather large standard deviations (Figure 12), which reflect that the entrainment rate of dense material varies significantly with time.

[48] We then compute a series of cases which are similar to case ET1 except that the buoyancy number B for these cases varies between 1.0 and 0.6. Figures 10b and 10d show the mesh distribution, nondimensional temperature and chemical composition for the case with $B = 0.6$ (i.e., the smallest B considered in here) after it reaches a steady state. It can be seen that a smaller buoyancy

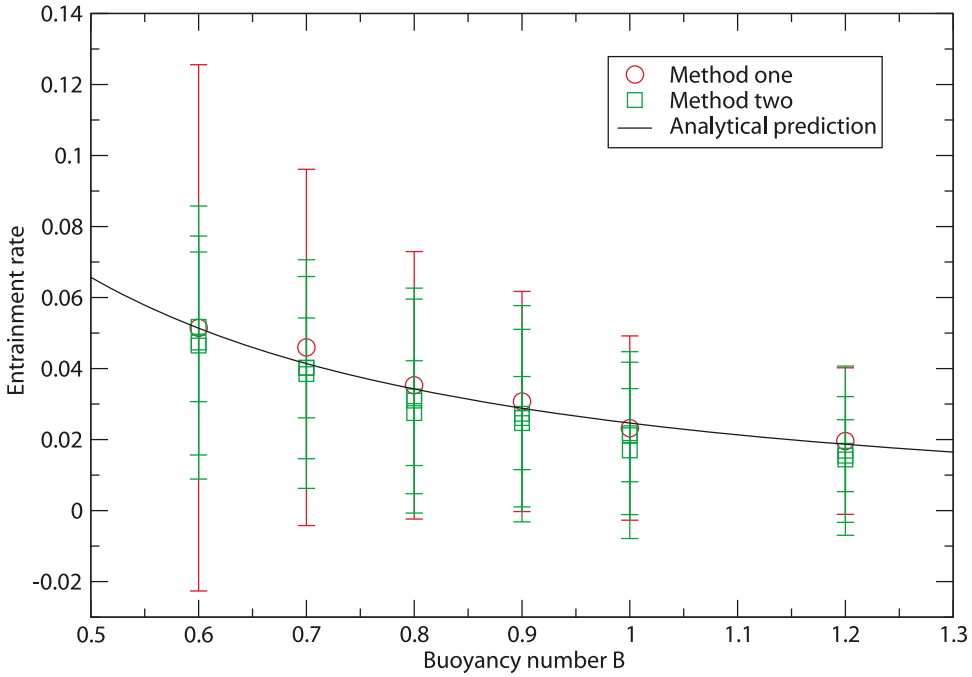

Figure 12. The time-averaged entrainment rates computed from two different methods for thermochemical convection cases with different chemical buoyancy number B. The red circles show the results from the first method, and the green squares show the integrated chemical flux computed across the three lines in Figure 10c and 10d from the second method. See text for the method details. The error bars show the standard deviations over the computed time interval. The black line shows the analytic prediction from the equation (17).

number leads to large deformation of the dense layer [*Davaille*, 1999]. Figure 11a shows the total composition with time for all these cases, suggesting that the conservation of chemical composition is well satisfied for these cases. Figure 11b shows the total volume of the entrained dense material above $z = 0.5$ with time for all these cases. We consider all these cases reaches steady state after $t = 0.25$ and compute the averaged entrainment rate thereafter. Figure 12 shows the resulted entrainment rate from the two different methods for all these cases. The results from these two methods match each other quite closely (Figure 12), and we thus consider our results as robust.

[49] *Davaille* [1999] showed that the entrainment rate is dependent on the buoyancy number if other parameters are fixed:

$$Q_m = cB^{-2}\frac{1}{1+1/B}, \tag{17}$$

where Q_m is the entrainment rate and c is a constant. Notice that equation (17) is a simplified form from *Davaille*'s [1999] since we are using constant viscosity and a fixed Rayleigh number here. In Figure 12, we plot the analytic prediction of entrainment rate according to equation (17). It can be observed that the analytic prediction and the numerical results agree with each other quite well. The constant c in the equation (17) is chosen to best fit the data, which means that varying c leads to vertical shifts of the analytic curve.

5. Conclusion

[50] In summary, we develop a 2-D finite element code with the adaptive mesh refinement techniques for compressible thermochemical mantle convection modeling. We use a quadtree data structure to facilitate the dynamical mesh refinement and coarsening. The stabilized $Q_1 - Q_1$ finite element is used to discretize the compressible thermochemical convection equations. We also implement an improved tracer method to better simulate the chemical compositions for thermochemical convection. The numerical results from our new code are compared to analytic solutions and benchmark results previously published. The agreement shows that our code provides accurate results with a smaller number of elements and shorter computational time compared with results from uniform meshes. We test the resolving capability of our code with the entrainment problem and compute the entrainment

rate of a dense layer by upwelling plumes. The agreement of the results from different methods and from the analytic predictions demonstrates that our code can be employed to study the thermochemical convection problems which require high spatial resolution to resolve chemical boundaries.

Acknowledgments

[51] We thank Georg Stadler for his technical help and suggestive comments and Peter van Keken for providing the benchmark data of the Rayleigh-Taylor problem. We also thank Rhodri Davies and Dave May for their constructive and detailed reviews that significantly improved this paper. This study is supported by National Science Foundation OCI-0749045 and David and Lucile Packard Foundation.

References

Abel, J. F., and M. S. Shephard (1979), An algorithm for multipoint constraints in finite element analysis, *Int. J. Numer. Methods Eng.*, *14*, 464–467, doi:10.1002/nme.1620140312.

Ainsworth, M., and J. T. Oden (2000), *A Posteriori Error Estimation in Finite Element Analysis*, 264 pp., Wiley, New York.

Albers, M. (2000), A local mesh refinement multigrid method for 3-D convection problems with strongly variable viscosity, *J. Comput. Phys.*, *160*, 126–150, doi:10.1006/jcph.2000.6438.

Babuvška, I., and W. C. Rheinboldt (1978), Error estimates for adaptive finite element computations, *SIAM J. Numer. Anal.*, *15*, 736–754, doi:10.1137/0715049.

Bank, R. E., A. H. Sherman, and A. Weiser (1983), Refinement algorithms and data structures for regular mesh refinement, in *Scientific Computing*, *IMACS Trans. Sci. Comput.*, vol. 1, edited by R. S. Stepleman et al., pp. 3–17, North-Holland, Amsterdam.

Blankenbach, B., et al. (1989), A benchmark comparison for mantle convection codes, *Geophys. J. Int.*, *98*, 23–38, doi:10.1111/j.1365-246X.1989.tb05511.x.

Braun, J., C. Thieulot, P. Fullsack, M. DeKool, C. Beaumont, and R. Huismans (2008), DOUAR: A new three-dimensional creeping flow numerical model for the solution of geological problems, *Phys. Earth Planet. Inter.*, *171*, 76–91, doi:10.1016/j.pepi.2008.05.003.

Brooks, A. N., and T. J. R. Hughes (1982), Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Comput. Methods Appl. Mech. Eng.*, *32*, 199–259, doi:10.1016/0045-7825(82)90071-8.

Bunge, H.-P., M. A. Richards, and J. R. Baumgardner (1997), A sensitivity study of 3-D spherical mantle convection at 10^8 Rayleigh number: Effects of depth dependent viscosity, heating mode and an endothermic phase change, *J. Geophys. Res.*, *102*, 11,991–12,007, doi:10.1029/96JB03806.

Burstedde, C., O. Ghattas, M. Gurnis, G. Stadler, E. Tan, T. Tu, L. Wilcox, and S. J. Zhong (2008), Scalable parallel mantle convection simulation on petascale computers, in *International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–15, Inst. of Electr. and Electr. Eng., Piscataway, N. Y., doi:10.1109/SC.2008.5214248.

Burstedde, C., O. Ghattas, G. Stadler, T. Tu, and L. C. Wilcox (2009), Parallel scalable adjoint-based adaptive solution of variable-viscosity Stokes flow problems, *Comput. Methods Appl. Mech. Eng.*, *198*, 1691–1700, doi:10.1016/j.cma.2008.12.015.

Davaille, A. (1999), Simultaneous generation of hotspots and superswells by convection in a heterogeneous planetary mantle, *Nature*, *402*, 756–760, doi:10.1038/45461.

Davies, D. R., J. H. Davies, O. Hassan, K. Morgan, and P. Nithiarasu (2007), Investigations into the applicability of adaptive finite element methods to two-dimensional infinite Prandtl number thermal and thermochemical convection, *Geochem. Geophys. Geosyst.*, *8*, Q05010, doi:10.1029/2006GC001470.

Davies, D. R., J. H. Davies, O. Hassan, K. Morgan, and P. Nithiarasu (2008) Adaptive finite element methods in geodynamics: Convection dominated mid-ocean ridge and subduction zone simulations, *Int. J. Numer. Methods Heat Fluid Flow*, *18*(7–8), 1015–1035, doi:10.1108/09615530810899079.

Dohrmann, C. R., and P. B. Bochev (2004), A stabilized finite element method for the Stokes problem based on polynomial pressure projections, *Int. J. Numer. Methods Fluids*, *46*, 183–201, doi:10.1002/fld.752.

Elman, H. C., D. J. Silvester, and A. J. Wathen (2005), *Finite Elements and Fast Iterative Solver With Applications in Incompressible Fluid Dynamics*, 400 pp., Oxford Univ. Press, Oxford, N. Y.

Enright, D., R. Fedkiw, J. Ferziger, and I. Mitchell (2002), A hybrid particle level set method for improved interface capturing, *J. Comput. Phys.*, *183*, 83–116, doi:10.1006/jcph.2002.7166.

Gonnermann, H. M., M. Manga, and A. M. Jellinek (2002), Dynamics and longevity of an initially stratified mantle, *Geophys. Res. Lett.*, *29*(10), 1399, doi:10.1029/2002GL014851.

Hager, B. H., and R. J. O'Connell (1979), Kinematic models of large-scale flow in the Earth's mantle, *J. Geophys. Res.*, *84*, 1031–1048, doi:10.1029/JB084iB03p01031.

Hieber, S. E., and P. Koumoutsakos (2005), A Lagrangian particle level set method, *J. Comput. Phys.*, *210*, 342–367, doi:10.1016/j.jcp.2005.04.013.

Huang, J. S. (2008), Controls on entrainment of a dense chemical layer by thermal plumes, *Phys. Earth Planet. Inter.*, *166*, 175–187, doi:10.1016/j.pepi.2008.01.006.

Hughes, T. J. R. (2000), *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, 683 pp., Dover, Mineola, N. Y.

Jellinek, A. M., and M. Manga (2002), The influence of a chemical boundary layer on the fixity, spacing and lifetime of mantle plumes, *Nature*, *418*, 760–763, doi:10.1038/nature00979.

Kellogg, L. H., B. H. Hager, and R. van der Hilst (1999), Compositional stratification in the deep mantle, *Science*, *283*, 1881–1884, doi:10.1126/science.283.5409.1881.

King, S. D., A. Raefsky, and B. H. Hager (1990), Conman: Vectorizing a finite element code for incompressible two-dimensional convection in the Earth's mantle, *Phys. Earth Planet. Inter.*, *59*, 195–207, doi:10.1016/0031-9201(90)90225-M.

King, S. D., C. Y. Lee, P. E. van Keken, W. Leng, S. J. Zhong, E. Tan, N. Tosi, and M. C. Kameyama (2010), A community benchmark for 2D Cartesian compressible convection in the Earth's mantle, *Geophys. J. Int.*, *180*, 73–87, doi:10.1111/j.1365-246X.2009.04413.x.

Lenardic, A., and W. M. Kaula (1993), A numerical treatment of geodynamic viscous flow problems involving the advection of

material interfaces, *J. Geophys. Res.*, *98*, 8243–8260, doi:10.1029/92JB02858.

Leng, W., and S. J. Zhong (2008), Viscous heating, adiabatic heating and energetic consistency in compressible mantle convection, *Geophys. J. Int.*, *173*, 693–702, doi:10.1111/j.1365-246X.2008.03745.x.

Löhner, R., K. Morgan, and O. C. Zienkiewicz (1985), An adaptive finite-element method for high-speed compressible flow, *Lect. Notes Phys.*, *218*, 388–392, doi:10.1007/3-540-13917-6_168.

McNamara, A. K., and S. J. Zhong (2004), Thermochemical structures within a spherical mantle: Superplumes or piles?, *J. Geophys. Res.*, *109*, B07402, doi:10.1029/2003JB002847.

McNamara, A. K., and S. J. Zhong (2005), Thermochemical structures beneath Africa and the Pacific Ocean, *Nature*, *437*, 1136–1139, doi:10.1038/nature04066.

Moresi, L., S. J. Zhong, and M. Gurnis (1996), The accuracy of finite element solutions of Stokes' flow with strongly varying viscosity, *Phys. Earth Planet. Inter.*, *97*, 83–94, doi:10.1016/0031-9201(96)03163-9.

Morton, G. M. (1966), A computer oriented geodetic data base and a new technique in file sequencing, technical report, Int. Bus. Mach. Corp., Ottawa, Ont., Canada.

Peraire, J., M. Vahdati, K. Morgan, and O. C. Zienkiewicz (1987), Adaptive remeshing for compressible flow computations, *J. Comput. Phys.*, *72*, 449–466, doi:10.1016/0021-9991(87)90093-3.

Samuel, H., and M. Evonuk (2010), Modeling advection in geophysical flows with particle level sets, *Geochem. Geophys. Geosyst.*, *11*, Q08020, doi:10.1029/2010GC003081.

Shephard, M. S. (1984), Linear multipoint constraints applied via transformation as part of a direct stiffness assembly process, *Int. J. Numer. Methods Eng.*, *20*, 2107–2112, doi:10.1002/nme.1620201112.

Sleep, N. H. (1988), Gradual entrainment of a chemical layer at the base of the mantle by overlying convection, *Geophys. J. Int.*, *95*, 437–447, doi:10.1111/j.1365-246X.1988.tb06695.x.

Stadler, G., M. Gurnis, C. Burstedde, L. C. Wilcox, L. Alisic, and O. Ghattas (2010), The dynamics of plate tectonics and mantle flow: From local to global scales, *Science*, *329*, 1033–1038, doi:10.1126/science.1191223.

Tackley, P. J. (1996), Effects of strongly variable viscosity on three-dimensional compressible convection in planetary mantles, *J. Geophys. Res.*, *101*, 3311–3332, doi:10.1029/95JB03211.

Tackley, P. J., and S. D. King (2003), Testing the tracer ratio method for modeling active compositional fields in mantle convection simulations, *Geochem. Geophys. Geosyst.*, *4*(4), 8302, doi:10.1029/2001GC000214.

Tu, T., D. R. O'Hallaron, and O. Ghattas (2005) Scalable parallel octree meshing for terascale applications, paper presented at International Conference for High Performance Computing, Networking, Storage, and Analysis, Seattle, Wash.

Tu, T., H. Yu, L. Ramirez-Guzman, J. Bielak, O. Ghattas, K. Ma, and D. R. O'Hallaron (2006) From mesh generation to scientific visualization: An end-to-end approach to parallel super computing, paper presented at International Conference for High Performance Computing, Networking, Storage, and Analysis, Tampa, Fla.

van Keken, P. E., S. D. King, H. Schmeling, U. R. Christensen, D. Neumeister, and M.-P. Doin (1997), A comparison of methods for the modeling of thermochemical convection, *J. Geophys. Res.*, *102*, 22,477–22,495, doi:10.1029/97JB01353.

Verfurth, R. (1996), *A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, 134 pp., Wiley-Teubner, Chichester, N. Y.

Zhong, S. J. (2005), Dynamics of Thermal plumes in 3D isoviscous thermal convection, *Geophys. J. Int.*, *162*, 289–300, doi:10.1111/j.1365-246X.2005.02633.x.

Zhong, S. J., and B. H. Hager (2003), Entrainment of a dense layer by thermal plumes, *Geophys. J. Int.*, *154*, 666–676, doi:10.1046/j.1365-246X.2003.01988.x.

Zhong, S. J., M. T. Zuber, L. Moresi, and M. Gurnis (2000), Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection, *J. Geophys. Res.*, *105*, 11,063–11,082, doi:10.1029/2000JB900003.